IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Context-aware Human Activity Recognition

Roghayeh Mojarad[1], Ferhat Attal[1], Abdelghani Chibani[1], Yacine Amirat[1]

Abstract— **One of the main challenges in designing Ambient Assisted Living (AAL) systems is Human Activity Recognition (HAR). The latter is crucial to improve the quality of people's lives in terms of autonomy, safety, and well-being. In this paper, a novel framework exploiting the contextual information of human activities is proposed for HAR. The proposed framework allows detecting and correcting classification errors automatically. Machine-learning models are firstly used to recognize human activities. These models may predict erroneous activities; therefore, detecting and correcting these errors is necessary to improve HAR. For this purpose, two Bayesian networks are used for classification error detection and classification error correction. The proposed framework is evaluated in terms of precision, recall, F-measure, and accuracy on the *Opportunity dataset*, a well-known dataset for multi-label human daily living activity recognition. The evaluation results demonstrate the ability of the proposed framework to improve HAR performance.**

I. Introduction

One of the main objectives of Ambient Assisted Living (AAL) systems is to proactively provide intelligent services to improve the quality of people's lives in terms of autonomy, safety, and well-being. Designing AAL systems that can autonomously monitor human's activities and provide assistance services poses several challenges of which Human Activity Recognition (HAR) which is critically important to adapt the assistance services to the user. HAR is becoming an active and a challenging research topic in several application domains including ambient assisted living [1], rehabilitation [2], and healthcare [3]. The goal of HAR is to recognize human daily activities from data collected through sensors worn by the user or disseminated in the environment. Data used to train classifiers to predict human activities are rarely ideal; as the data could be mislabeled, highly variant, unbalanced, noisy, and with an unclear line of demarcation; moreover, the sensor reading could be erroneous or lost.

A practical approach to enhance HAR performance in terms of accuracy is multi-label classification, which is an extension of traditional multi-class classification. The multi-label classification is based on the assumption that a data sample can be marked with more than one label [4]. One of the main problems in multi-label classification is how to model and exploit the relationships between the labels to prevent classification errors. In the multi-label classification, the relationships between labels can be considered as context. According to A.K. Dey [5], a context is any information that can be used to characterize the situation of an entity; the latter is a person, place, or an object that is considered relevant to the interaction between a user and an application. The relationships between the labels can be used to obtain efficient models for automatically detecting and correcting classification errors. To the best of our knowledge, there is no existing classification error detection and correction approach in the literature in multi-label HAR domain. Therefore, this study is the first attempt to deal with this issue.

In this paper, a robust multi-label classification framework with classification error detection and correction is proposed to recognize human activities. Multi-labeling in the domain of HAR allows for a more comprehensive and finer description of activities, where each activity can be described as a combination of sub-activities while considering its execution environment, the used objects, the body gesture of the person executing the activity, and other concepts. This framework is driven by the idea that the best HAR performance cannot be achieved without modeling and exploiting statistical relationships between the labels corresponding to the context of the activity such as the manipulated objects, gesture, posture, place, etc. It is widely recognized that contextual information can be beneficial for enhancing the capabilities of AAL systems. The proposed framework uses two modules to process sensors data: (i) classification module and (ii) classification error detection and correction module. In the first module, machine-learning models are used to recognize human activities by predicting a set of context labels describing the ongoing activities. In this module, trained models are used to classify context labels separately. These models may lead to classification errors in the HAR. To enhance the HAR performance, the second module is proposed to detect and correct classification errors using two Bayesian Networks that allow modeling the statistical relationships that exist between the labels assigned to each data sample. An error is thus detected when the existing relationships between the predicted labels describing the ongoing activities fail to match with any of the existing paths in the Bayesian network dedicated for the error detection. The detected error is then corrected heuristically by finding the most appropriate relationship between the labels in the Bayesian network dedicated to the correction. The main contribution of this paper is proposing a multi-label classification framework with the ability of classification error detection and correction. In order to demonstrate the effectiveness of the proposed framework, performance analysis has been conducted by considering 15 representative attempts of the state-of-the-art that treats HAR based on standard baseline hierarchical multi-label activity labels. The performance analysis was conducted on

[1] All authors are with Univ Paris Est Creteil, LISSI, F-94400 Vitry, France.
roghayeh.mojarad@u-pec.fr, ferhat.attal@u-pec.fr, abdelghani.chibani@u-pec.fr, amirat@u-pec.fr

the opportunity dataset [6], which is a unique hierarchical multi-label HAR dataset in contrast to the others in the literature (e.g., MHEALTH [7], PAMAP2 [8], USC-HAD [9], UTD-MHAD [10], WHARF [11], and WISDM [12]), which are not multi-label datasets. The analysis results show the significant improvements using the proposed framework over machine-learning models.

II. Related Works

HAR has been widely studied in the literature in the last years. The proposed classification approaches can be classified into two main categories [13], [14]: (i) conventional classification approaches such as Naïve Bayes (NB), K-Nearest Neighbors (K-NN), Support Vector Machine (SVM), Random Forests (RFs), K-Means, Gaussian Mixture Model (GMM), Decision Tree (DT), and Hidden Markov Model (HMM) and (ii) deep learning classification approaches such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Deep Belief Network (DBN), Restricted Boltzmann Machines (RBM), and Long Short-Term Memory (LSTM). Research on HAR is increasingly focusing on multi-label classification approaches [4], [15], which belong to both the aforementioned categories.

In multi-label classification approaches, a data sample of the features vector can belong to more than one label [16]. In general, there are two main categories of multi-label classification approaches [4]: (i) Problem Transformation (PT) approaches and (ii) Algorithm Adaptation (AA) approaches [16]. In the first category, the multi-label problem is transferred into other well-established learning scenarios [17], [18]. The well-known PT approaches include Binary Relevance (BR) [19] and Classifier Chains (CC) [20]. In these approaches, the multi-label classification problem is transformed into a binary classification problem. Another well-known PT approach is the calibrated label ranking approach [21] which transforms the multi-label classification problem into the label ranking problem. Random k-labelsets approach [22] is another well-known PT approach which transforms the multi-label classification problem into the multi-class classification problem. The second category of multi-label classification approaches, i.e., AA approaches [23], [24] adapt a popular classification model to deal with multi-label data directly. ML-KNN adapts a lazy learning model [25] whereas Rank-SVM adapts a kernel learning model [26]. In the AA approach presented in [24], a general conditional dependency network model is used to provide a multi-label classification. In [15], a low-rank formulation for weakly supervised classification is proposed to recognize multi-label human activities. To sum up, the main idea of PT approaches is to fit data to the classification approach while the main idea of AA approaches is to fit the classification approach to data [16].

In this paper, a classification error detection and correction approach is proposed to improve the accuracy and robustness of HAR as an important challenge in AALs. There are some works that deal with handling sensor errors in HAR. In [27], an exercise recognition based on accelerometer and gyroscope is presented. This study focuses only on the filtering of undesirable raw data using a k-wave filter. In [28], the authors propose a semantic reasoning approach for activity recognition by taking into account uncertain data due to sensor errors such as hardware failures, energy depletion, and communication problems. The uncertainty is translated into a confidence value for each current event. The confidence value is measured based on the hardware characteristics and operational properties of each sensor. The proposed context-aware reasoning uses this confidence to make more accurate activity recognition. This study does not deal with classification errors. Another limitation of this approach is its dependency on the sensing technologies hardware and operational characteristics. The sensor errors have also been studied outside of the HAR domain [29], [30]. For example, in [30], two approaches have been proposed based on Markov and Stide models to detect and correct errors. In the Markov-based model, a transition matrix is constructed. When the probability of transition from the previous event to the current event does not reach a predefined threshold, an error is detected. In the Stide-based model, a list of events and their frequencies is constructed. Afterward, if the frequency of an unmatched event exceeds the threshold, an error is detected. In order to correct the detected error, the defined constraints for each erroneous event are examined to find the source of error and a suitable way to correct that event. The constraints of each erroneous event are based on the sequences of sub-events of that event. The number of constraints is related to the length of the window in the correction algorithm. These approaches depend strongly on data sequences and do not consider the context of events.

Several studies on multi-label HAR are proposed in the literature but none of them exploit the multi-label aspect to detect and correct classification errors. In this study, to enhance the HAR performance, the context of human activities is considered by modeling the statistical relationships that exist between the labels assigned to each data sample.

III. Human activity recognition framework

Fig. 1 shows the general architecture of the proposed framework. This architecture is composed of two main modules: (i) classification module and (ii) classification error detection and correction module. The first module consists of machine-learning models in order to classify the input data, which are multi-label data, into suitable activities. Each activity can be described using a set of labels according to different annotation levels, where each level provides specific characteristics of the activity such as *high-level activity* and *low-level activity*. For example, the *Cleanup* activity is a *high-level activity* while the *Open* activity is a *low-level activity*. In the classification module, the number of used machine-learning models corresponds to the number of annotation levels of activities. In Fig. 1, l represents the number of annotation levels of activities in the multi-label activity dataset; therefore, l machine-learning models are trained independently. Since the relationships between labels, which are annotation levels, are not taken into account,

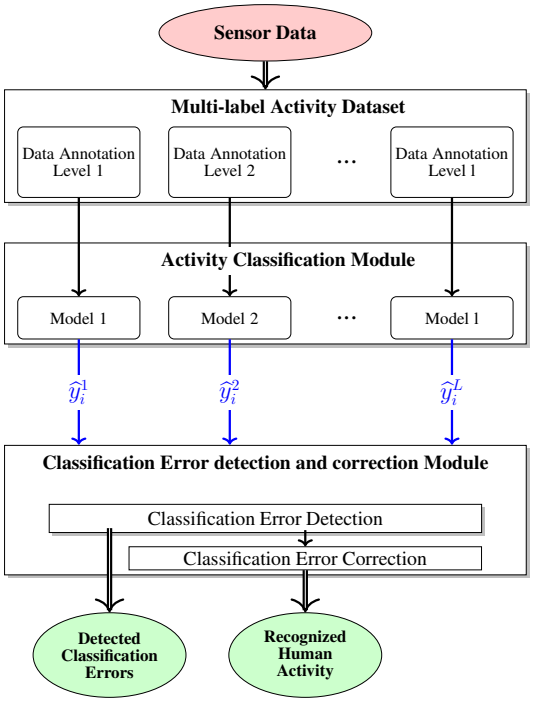

Fig. 1. Architecture of the proposed framework.

the predicted labels obtained from the activity classification module may be erroneous. Therefore, the second module is introduced to detect and correct the erroneous labels predicted by analyzing the relationships between the labels.

A. Multi-label Activity Dataset

In Fig. 1, the first component of the proposed architecture is the multi-label activity dataset, which includes sensors data, activity, and context labels. The multi-label input data can be used to represent the different annotation levels of activities. The multi-label input data D can be represented as a set of pairs formed from the data X_i and a vector of labels Y_i:

$$D = \{(X_i, Y_i) \mid 1 \leq i \leq N\} \tag{1}$$

where $i \in \{1, 2, ..., N\}$; N represents the total number of data samples. X_i represents the i^{th} data sample. Each data sample is composed of d data features. A vector of labels Y_i is assigned to the data sample i:

$$X_i = \{x_i^1, x_i^2, ..., x_i^m, ..., x_i^d\} \tag{2}$$

$$Y_i = \{y_i^1, y_i^2, ..., y_i^k, ..., y_i^l\} \tag{3}$$

where l represents the number of labels that are assigned to each data sample and y_i^k represents the k^{th} label assigned to the data sample i. Each label is selected from a specific number of labels. For instance, the first label is selected from t labels while the second label is selected from o labels, where t and o are positive values that may not be equal:

$$y_i^1 \in \{c_1^1, c_2^1, ..., c_t^1\} \tag{4}$$
$$y_i^2 \in \{c_1^2, c_2^2, ..., c_o^2\}$$

B. Activity Classification Module

In the classification module, input data are classified into different labels independently. The classification module can be formalized as a function f such as:

$$\widehat{Y}_i = f(X_i) \tag{5}$$

where $\widehat{Y}_i$ is the vector of the predicted labels such that $\widehat{Y}_i = \{\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l\}$; $\widehat{y}_i^1$, $\widehat{y}_i^2$, and $\widehat{y}_i^l$ are respectively the outputs of machine-learning *model 1*, *model 2*, and *model l*. The function f can be any supervised machine-learning model including classification approaches such as NB, SVM, RFs, DT, CNN or LSTM, which may predict labels wrongly.

C. Classification Error Detection and Correction Module

1) Classification error detection: To detect classification errors, the relationships between labels should be modeled. Graphical models such as Markov network and Bayesian network are commonly used in the literature to model the dependencies or relationships between variables. In this study, variables correspond to labels predicted by the activity classification module. A Markov network is an undirected probabilistic graphical model while a Bayesian network is a directed acyclic probabilistic graphical model. Each node in the directed graph corresponds to a random variable and each directed edge represents a probabilistic dependency. If there is a directed edge from node A to node B, it means that node A is a parent of node B. In the probability theory and statistics, the conditional probability distribution of two variables A and B is the probability distribution of B when A is known to be a particular value; in some cases, the conditional probabilities may be represented as functions containing the unspecified value a of A as a parameter. In the case of undirected graphical models, there is no direction and hence no natural conditioning; therefore, using conditional probability seems erratic. Moreover, the learning problem associated with a Markov network is much more challenging than their counterparts in a Bayesian network [24]. Therefore, in this study, a Bayesian network is used to model the relationships between labels.

Bayesian network is represented as Directed Acyclic Graph (DAG), G = (V, E), with a set of conditional probability tables, where V represents a set of nodes and E represents a set of directed edges. Each node of the network models a conditional probability distribution given its parents in the network. The learning task of the Bayesian network can be categorized into two subtasks: (i) structure learning and (ii) parameter learning. The structure learning identifies the topology of the network while the parameter learning determines the numerical parameters such as conditional probabilities for a given network topology. In the proposed framework, the K2 algorithm is used for the structure learning [31] and the *Bayesian method* is used for parameter learning [32]. In the case of existing a directed edge from node y_i^1 to node y_i^2, with $y_i^1, y_i^2 \in G$, node y_i^1 is a parent

of node y_i^2. The joint probability distribution among nodes is represented as follows:

$$P(y_i^1, y_i^2, ..., y_i^l) = \prod_{v=1}^{l} P(y_i^v | Parent(y_i^v)) \quad (6)$$

The conditional probability between variables is calculated as follows:

$$\begin{aligned} P(y_i^v | Parent(y_i^v)) &= \frac{P(y_i^v, Parent(y_i^v))}{P(Parent(y_i^v))} \\ &= \alpha P(y_i^v, Parent(y_i^v)) \end{aligned} \quad (7)$$

where α is a normalization constant that can be calculated as follows:

$$\begin{aligned} \alpha &= \frac{1}{P(Parent(y_i^v))} \\ &= \frac{1}{P(y_i^v, Parent(y_i^v)) + P(\neg y_i^v, Parent(y_i^v))} \\ &= \frac{1}{P(y_i^1, y_i^2, ..., y_i^v, ..., y_i^l) + P(y_i^1, y_i^2, ..., \neg y_i^v, ..., y_i^l)} \end{aligned} \quad (8)$$

If the relationship between two labels fails to match with any of the existing paths of the Bayesian network, an error is detected. The proposed error detection module is formalized as follows:

$$DetectedError = 1 - P(\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l) \quad (9)$$

It means that if the joint probability among labels is equal to zero, then the value of $DetectedError$ is equal to one; consequently, the module will detect an error. In other words, the error detection algorithm detects an error when the relationships between the predicted labels describing the ongoing activities do not exist in the Bayesian network. The pseudocode of the error detection algorithm is provided in Algorithm 1. The input of this algorithm corresponds to the output of the classification module known as an event Ev, which is composed of l labels. The joint probability of labels is then calculated (lines 4, 5, 6); if this probability is equal to zero (line 7), then an error is detected (line 8) and the event Ev is considered as an erroneous event (line 9). Otherwise (line 10), no error is detected (lines 11, 12). In the worst case, the complexity of the error detection algorithm is $O(\prod_{i=1}^{l} K_i)$, where l represents the number of models and K_i represents the number of labels for model i.

2) Classification error correction: To correct the detected errors, the context of human activities should be considered through the relationships between correct labels of activities and labels of the detected erroneous event using a Bayesian network. The proposed error correction module can be formalized as a function g:

$$Z_i = g(\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l) \quad (10)$$

where $\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l$ are labels of detected erroneous event predicted by the machine-learning models. Z_i is the corrected event such that $Z_i = \{z_i^1, z_i^2, ..., z_i^l\}$; z_i^1, z_i^2, and z_i^l are respectively corrected labels of $\widehat{y}_i^1, \widehat{y}_i^2, ...,$ and $\widehat{y}_i^l$. The pseudocode of the classification error correction is shown in Algorithm 2. The inputs of this algorithm correspond to the outputs of the Algorithm 1: the detected error (DetectedError) and the erroneous event (EvwCE). In Algorithm 2, the probable correct events are selected considering the existing relationships between labels of the erroneous event and existing paths in the Bayesian network (line 1). If the number of probable correct events is greater than one (line 2), then the most appropriate one is heuristically chosen based on the frequency of these events and the number of labels required to be changed (line 3). Otherwise (line 4), there is only one probable correct event that is considered as an event without classification error ($EvwoCE$). The complexity of the error correction algorithm is similar to that of the error detection algorithm, which is $O(\prod_{i=1}^{l} K_i)$. An experiment was recorded to show how the proposed framework detects and corrects classification errors[1].

Algorithm 1 Pseudocode of the error detection

Input: an Event with l labels ($Ev(\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l)$)
Output: an variable of detected error ($DetectedError$); an Event with a Classification Error ($EvwCE$)

1: DetectedError ← False *# Initialization*
2: $EvwCE$ ← NULL
3: tmp ← 0
4: FOR v in range (1, l): *# Joint probability calculation*
5: $tmp \leftarrow tmp * P(y_i^v | Parent(y_i^v))$
6: $P(Ev(\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l)) \leftarrow tmp$
 # Check to detect error
7: $IF(P(Ev(\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l)) == 0)$: *# Error is detected*
8: $DetectedError \leftarrow$ True
9: $EvwCE \leftarrow Ev(\widehat{y}_i^1, \widehat{y}_i^2, ..., \widehat{y}_i^l)$
10: ELSE: *# No error is detected*
11: $DetectedError \leftarrow$ False
12: $EvwCE \leftarrow$ NULL
13: **return** $DetectedError$, $EvwCE$

Algorithm 2 Pseudocode of the error correction

Input: the variable of detected error ($DetectedError$); an Event with a Classification Error ($EvwCE$)
Output: an Event without a Classification Error ($EvwoCE$)

1: $AEvwoCE \leftarrow$ find probable correct event (BN model, $EvwCE$) # Check the number of probable correct event
2: IF Size($AEvwoCE$) > 1 :
 # There is more than one probable correct event
3: $EvwoCE \leftarrow$ find suitable correct event($AEvwoCE$, BN model, $EvwCE$)
4: ELSE: *# There is only one probable correct event*
5: $EvwoCE \leftarrow AEvwoCE$
6: **return** $EvwoCE$

[1] http://lissi.fr/videos/ErrorDetectionAndCorrectionVideo.mp4

IV. Results and discussion

In this section, the proposed framework is evaluated in terms of precision, recall, *F*-measure, and accuracy. Different machine-learning models have been evaluated to show the effectiveness of the proposed framework in the performance improvements of these machine-learning models. Moreover, the proposed framework is evaluated in comparison with 15 baseline approaches using *Opportunity dataset* [6], a benchmark and a unique dataset for multi-label human daily living activity recognition.

A. Description of the dataset

Opportunity dataset includes data collected from 72 sensors (wearable sensors, object sensors, and ambient sensors). Four volunteer human subjects were involved in this dataset such that each participant repeated each activity six times. The activities were annotated in seven annotation levels: (1) *high-level-activity*, (2) *middle-level-activity*, (3) *locomotion*, (4) *right-hand-activity*, (5) *right-hand-object*, (6) *left-hand-activity*, and (7) *left-hand-object*. The *locomotion* annotation level illustrates the general situation of the activity. Each *high-level-activity* annotation level is composed of multiple *middle-level-activity* annotation levels. For instance, *Cleanup* is a label of *high-level-activity* annotation level because *Open Dishwasher* and *Clean Table*, which are the labels of *middle-level-activity* annotation level, are sub-activities of *Cleanup*. The *left-hand-activity* and *right-hand-activity* annotation levels represent the activities performed by the user's left hand and right hand, respectively. These types of activities are classified as *low-level-activity* annotation level. Since an activity usually uses an object and activities can be performed by the user's left hand and right hand, objects can be categorized into two groups including *left-hand-object* and *right-hand-object* annotation levels.

The number of labels for the *locomotion*, *low-level-activity*, *middle-level-activity*, *high-level-activity*, and *left-hand-object* or *right-hand-object* annotation levels are 4, 13, 17, 5, and 23; see Table I. The total recorded time was approximately 268 hours. The sampling frequency rate of recording is 30 Hz while the total number of recorded data points is 28,976,744.

TABLE I

The labels of each annotation level in *Opportunity Dataset*.

Levels	**Labels**
Locomotion	Stand, Walk, Sit, Lie
High-Level-Activity	Relaxing, Coffee Time, Early Morning, Cleanup, Sandwich Time
Middle-Level-Activity	Open Door1, Open Door2, Close Door1, Close Door2, Open Fridge, Close Fridge, Open Dishwasher, Close Dishwasher, Open Drawer1, Close Drawer1, Open Drawer2, Close Drawer2, Open Drawer3, Close Drawer3, Clean Table, Drink from Cup, Toggle Switch
Left or Right - Hand-Activity	Unlock, Stir, Lock, Close, Reach, Open, Sip, Clean, Bite, Cut, Spread, Release, Move
Left or Right - Hand-Object	Bottle, Salami, Bread, Sugar, Dishwasher, Switch, Milk, Drawer3 (lower), Spoon, Knife Cheese, Drawer2 (middle), Table, Glass, Cheese, Chair, Door1, Door2, Plate, Drawer1 (top), Fridge, Cup, Knife Salami, Lazy chair

B. Results

To evaluate the added value of the proposed framework in terms of robustness with respect to classification errors, several machine-learning models including NB, DT, SVM, and RFs as well as Deep Residual Bidirectional Long Short-term Memory LSTM (Deep-Res-Bidir-LSTM) network [33] have been implemented in the classification module.

1) Performance of the activity classification module: machine-learning models have been evaluated in terms of precision, recall, *F*-measure, and accuracy. The results obtained using NB, DT, SVM, RFs, and Deep-Res-Bidir-LSTM are shown in Table II, Table III, Table IV, Table V, and Table VI, respectively. The RFs model yields the highest average F-measure, 81.90%, while for Deep-Res-Bidir-LSTM, the average F-measure is 81.13%. RFs and Deep-Res-Bidir-LSTM yield sensibly better performance in comparison with the other machine-learning models. In terms of F-measure, the most distinguishable performance difference between RFs and DT models is about 20% in the *middle-level-activity* annotation level. This can be explained by the fact that RFs is an ensemble of DTs learned by combining the bootstrap aggregating (bagging) method and using randomization in the selection of partitioning data nodes in the construction of DTs. The RFs model gives the best results in terms of precision, recall, *F*-measure, and accuracy.

2) Performance of the classification error detection and correction module: Table II shows the classification results obtained with the NB model and the proposed framework based on the NB model, which improves the NB model significantly, with an improvement of 20.24% in terms of *F*-measure on average. The maximum improvement takes place in the case of *right-hand-activity* annotation level in terms of *F*-measure. Table III shows the obtained results using the DT model and the proposed framework based on the DT model. One can observe an improvement up to 26.16% in the case of *right-hand-object* annotation level in terms of *F*-measure. The classification results obtained with the SVM model and the proposed framework based on the SVM model are shown in Table IV. A minimum improvement of 1.46% is observed in the *middle-level-activity* annotation level and a maximal improvement of 5.48% is obtained in the *locomotion* annotation level. Table V summarizes the classification results obtained using the RFs model and the proposed framework based on the RFs model, which shows a slight improvement over the RFs model (3.85%) in terms of *F*-measure. The maximum improvement is related to the *right-hand-object* annotation level. Table VI shows the classification results obtained with the Deep-Res-Bidir-LSTM model and the proposed framework based on the Deep-Res-Bidir-LSTM model. The average improvement obtained with the proposed framework in terms of *F*-measure is 5.45%. The best improvement in terms of *F*-measure and precision is related to the *right-hand-object* annotation level. As can be observed from the precision and recall values, in

TABLE II

NB MODEL IN COMPARISON WITH THE PROPOSED FRAMEWORK BASED ON NB.

Models	NB (%)				Proposed framework based on NB(%)			
	Precision	Recall	F-measure	Accuracy	Precision	Recall	F-measure	Accuracy
Locomotion	64.16	81.66	67.15	83.38	71.43	85.40	75.93	92.95
High-Level-Activity	71.06	78.63	72.66	86.99	84.13	87.63	85.18	93.22
Left-Hand-Activity	56.90	62.45	55.61	84.83	73.79	79.19	75.28	93.45
Left-Hand-Object	60.49	67.87	59.75	96.18	85.44	92.27	88.16	98.45
Right-Hand-Activity	49.17	42.99	40.13	79.12	69.56	72.71	69.94	92.95
Right-Hand-Object	63.36	56.83	57.47	95.05	79.44	74.75	75.81	97.83
Middle-Level-Activity	60.04	69.77	58.23	93.99	82.83	86.33	82.35	97.61
Average	60.74	65.74	58.71	88.50	78.08	82.61	78.95	95.20

TABLE III

DT MODEL IN COMPARISON WITH THE PROPOSED FRAMEWORK BASED ON DT.

Models	Decision tree(%)				Proposed framework based on DT(%)			
	Precision	Recall	F-measure	Accuracy	Precision	Recall	F-measure	Accuracy
Locomotion	76.17	71.69	73.77	87.02	86.20	87.05	86.37	92.48
High-Level-Activity	90.81	92.78	91.70	96.47	96.55	97.62	97.03	98.48
Left-Hand-Activity	70.05	71.86	70.59	89.70	84.84	85.93	85.25	95.91
Left-Hand-Object	91.12	89.14	88.30	99.11	95.91	98.41	99.90	99.90
Right-Hand-Activity	61.35	68.52	63.21	90.99	86.39	84.68	84.92	95.91
Right-Hand-Object	65.42	71.52	64.68	96.34	91.25	94.68	90.84	99.17
Middle-Level-Activity	72.12	79.74	73.55	96.23	97.03	97.49	97.14	98.94
Average	75.29	77.89	75.11	93.72	91.16	92.26	91.63	97.27

TABLE IV

SVM MODEL IN COMPARISON WITH THE PROPOSED FRAMEWORK BASED ON SVM.

Models	SVM (%)				Proposed framework based on SVM(%)			
	Precision	Recall	F-measure	Accuracy	Precision	Recall	F-measure	Accuracy
Locomotion	86.85	80.40	82.96	93.15	87.85	81.98	84.42	93.62
High-Level-Activity	95.33	90.43	92.48	95.79	97.19	95.77	96.36	97.77
Left-Hand-Activity	48.45	50.36	48.59	88.69	53.62	55.54	53.85	90.71
Left-Hand-Object	90.18	89.21	88.65	98.63	93.10	94.64	93.47	99.19
Right-Hand-Activity	55.65	48.18	50.28	90.02	60.23	53.69	55.63	91.85
Right-Hand-Object	81.53	85.91	81.07	97.99	85.52	90.87	85.57	98.63
Middle-Level-Activity	78.44	81.50	77.85	96.88	83.64	87.04	83.33	97.89
Average	76.63	75.14	74.55	94.45	80.16	79.93	78.94	95.66

TABLE V

RFS MODEL IN COMPARISON WITH THE PROPOSED FRAMEWORK BASED ON RFS.

Models	RFs(%)				Proposed framework based on RFs(%)			
	Precision	Recall	F-measure	Accuracy	Precision	Recall	F-measure	Accuracy
Locomotion	94.45	67.25	73.90	94.46	95.81	74.39	81.26	95.63
High-Level-Activity	92.63	86.95	88.45	93.95	91.61	91.28	90.15	94.29
Left-Hand-Activity	85.65	79.45	77.36	95.03	86.30	81.28	79.69	95.50
Left-Hand-Object	95.46	95.45	94.62	99.26	95.62	97.69	96.08	99.59
Right-Hand-Activity	82.10	71.83	74.49	92.21	83.90	76.65	78.12	93.48
Right-Hand-Object	70.60	72.00	71.08	98.48	79.12	80.64	79.80	98.96
Middle-Level-Activity	92.37	95.98	93.44	98.62	93.84	97.61	95.18	99.22
Average	87.60	81.37	81.90	96.00	89.45	85.65	85.75	96.66

this annotation level, the performance of the Deep-Res-Bidir-LSTM model is lower in comparison with other annotation levels. Therefore the number of errors in this annotation level is higher; hence, the classification error detection and correction module can better improve the performance of the Deep-Res-Bidir-LSTM model in this annotation level. It is worth noting that the improvements of the proposed framework depend on the relationships between the *right-hand-activity*, *right-hand-object*, *middle-level-activity*, *left-hand-activity*, *left-hand-object*, *locomotion*, and *high-level-activity* annotation levels. For instance, if the machine-learning models used for the *right-hand-object* and *right-hand-activity* annotation levels respectively report *Cup* and *Close*, the classification error detection and correction module can detect the error (*Close Cup*) because there is no path including *Close Cup* in the Bayesian network. Then, based on the labels of other annotation levels, the classification error detection and correction module finds the source of error and

TABLE VI

DEEP-RES-BIDIR-LSTM MODEL IN COMPARISON WITH THE PROPOSED FRAMEWORK BASED ON DEEP-RES-BIDIR-LSTM MODEL.

Models	Deep-Res-Bidir-LSTM(%)				Proposed framework based on Deep-Res-Bidir-LSTM(%)			
	Precision	Recall	F-measure	Accuracy	Precision	Recall	F-measure	Accuracy
Locomotion	89.51	80.91	84.99	89.46	94.48	89.42	91.88	94.43
High-Level-Activity	94.93	90.48	92.65	96.24	94.93	90.48	92.65	96.24
Left-Hand-Activity	78.95	71.85	75.23	91.33	83.91	76.37	79.96	92.98
Left-Hand-Object	95.88	94.98	95.43	99.49	96.87	95.49	96.18	99.57
Right-Hand-Activity	78.57	71.43	74.83	92.45	83.54	81.47	82.49	94.94
Right-Hand-Object	65.06	59.18	61.98	95.80	75.00	68.22	71.45	96.85
Middle-Level-Activity	84.99	80.77	82.83	97.65	89.96	92.96	91.44	98.88
Average	83.98	78.51	81.13	94.63	88.39	84.92	86.58	96.27

corrects that error. Consequently, in this example, the module improves the performance of the machine-learning model for the *right-hand-activity* or *right-hand-object* annotation level. Moreover, the proposed framework is independent of the used machine-learning model and the annotation level; but it depends on the number and diversity of the classification errors; i.e., it can best improve the performance of the machine-learning model for the annotation level with more numerous and diverse classification errors. For instance, the performance improvement obtained using the proposed framework based on SVM is lower than that obtained using the proposed framework based on DT. The classification errors obtained using SVM are not as huge and diverse as of the classification errors in DT. In summary, the obtained results show that the proposed framework improves the HAR using detecting and correcting the classification errors by considering the contextual information.

C. Baselines

With regard to the *Opportunity dataset*, 15 baseline approaches are compared with the proposed framework. The main baseline approaches are multi-view stacking classification approach [34], Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), K-NN using K=1 and K=3, and Nearest Centroid Classifier (NCC) [35]. Moreover, to show the efficiency of the proposed framework in comparison with deep-learning approaches, Autoencoder (AE), LSTM, CNN, Multi-Layer-Perceptron (MLP) [36] are considered as baseline approaches; Table VII shows the results in terms of accuracy. The proposed framework based on DT obtains the best performance. These results demonstrate that the proposed framework can remarkably improve machine-learning models. As in the proposed framework, the weaknesses of classifiers are exploited to train the classification error correction algorithm, better corrections and ultimately, better performance is obtained.

TABLE VII

COMPARISON WITH THE BASELINE APPROACHES.

Models	Accuracy(%)
Baseline Approaches [34]	
Accelerometer view	84.3
Gyroscope view	82.1
Magnetometer view	88.9
Aggregated views	91.4
Multi-View Stacking	92.5
Baseline Approaches [35]	
LDA	60
QDA	64
1-NN	82
3-NN	83
NCC	54
Baseline Approaches [36]	
AE	87.80
CNN	90.58
LSTM	91.29
MLP	91.11
Hybrid CNN-LSTM	91.76
Proposed framework	
based on RFs	96.66
based on DT	**97.27**
based on NB	95.20
based on SVM	95.66
based on Deep-Res-Bidir-LSTM	96.27

V. CONCLUSION AND FUTURE WORK

In this paper, a robust multi-label HAR framework is proposed in the context of AAL systems. Multi-labeling in the HAR domain allows for a comprehensive and finer description of activities. The proposed framework consists of two main modules: (i) classification module and (ii) classification error detection and correction module. The first module uses machine-learning models to classify the multi-label input data into suitable activity labels independently. The second module analyzes the relationships between predicted labels, the outputs of the machine-learning models to detect and correct classification errors. The evaluation results show that the proposed framework is generic and able to detect and correct classification errors of any multi-label classification approaches. The proposed framework significantly improves the performance of machine-learning models. Moreover, the proposed framework outperforms baseline approaches exploiting the same dataset. The obtained results show the robustness of the proposed framework against classification errors for HAR. In terms of research perspectives to this study, an interesting topic is to use Dynamic Bayesian Network (DBN) to take into account the temporal aspect of human activities. Moreover, another interesting research direction to explore is to exploit commonsense-reasoning to detect inconsistencies among labels predicted by machine-learning models in order to improve the HAR performance.

REFERENCES

[1] R. Mojarad, F. Attal, A. Chibani, S. R. Fiorini, and Y. Amirat, "Hybrid approach for human activity recognition by ubiquitous robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018, pp. 5660–5665.

[2] E. Kańtoch, "Human activity recognition for physical rehabilitation using wearable sensors fusion and artificial neural networks," in *Computing in Cardiology (CinC)*, Sep. 2017, pp. 1–4.

[3] H. F. Nweke, Y. W. Teh, M. A. Al-garadi, and U. R. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," *Expert Systems with Applications*, vol. 105, pp. 233–261, 2018.

[4] J. Huang, F. Qin, X. Zheng, Z. Cheng, Z. Yuan, W. Zhang, and Q. Huang, "Improving multi-label classification with missing labels by learning label-specific features," *Information Sciences*, vol. 492, pp. 124 – 146, 2019.

[5] A. K. Dey, "Understanding and using context," *Personal and Ubiquitous Computing*, vol. 5, no. 1, pp. 4–7, Feb 2001.

[6] R. Chavarriaga, H. Sagha, A. Calatroni, S. T. Digumarti, G. Tröster, J. del R. Millán, and D. Roggen, "The opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 2033 – 2042, 2013.

[7] O. Banos, R. Garcia, J. A. Holgado-Terriza, M. Damas, H. Pomares, I. Rojas, A. Saez, and C. Villalonga, "mhealthdroid: A novel framework for agile development of mobile health applications," in *Ambient Assisted Living and Daily Activities*, L. Pecchia, L. L. Chen, C. Nugent, and J. Bravo, Eds. Cham: Springer International Publishing, 2014, pp. 91–98.

[8] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," in *16th International Symposium on Wearable Computers*, June 2012, pp. 108–109.

[9] M. Zhang and A. A. Sawchuk, "Usc-had: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proceedings of the ACM Conference on Ubiquitous Computing*, ser. UbiComp '12. New York, NY, USA: ACM, 2012, pp. 1036–1043.

[10] C. Chen, R. Jafari, and N. Kehtarnavaz, "Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor," in *IEEE International Conference on Image Processing (ICIP)*, Sep. 2015, pp. 168–172.

[11] B. Bruno, F. Mastrogiovanni, and A. Sgorbissa, "A public domain dataset for adl recognition using wrist-placed accelerometers," in *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, Aug 2014, pp. 738–743.

[12] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *SIGKDD Explor. Newsl.*, vol. 12, no. 2, pp. 74–82, Mar. 2011.

[13] Y. Wang, S. Cang, and H. Yu, "A survey on wearable sensor modality centred human activity recognition in health care," *Expert Systems with Applications*, vol. 137, pp. 167 – 190, 2019.

[14] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical human activity recognition using wearable sensors," *Sensors*, vol. 15, no. 12, pp. 31 314–31 338, 2015.

[15] E. Adeli Mosabbeb, R. Cabral, F. De la Torre, and M. Fathy, "Multi-label discriminative weakly-supervised human activity recognition and localization," in *Computer Vision*, D. Cremers, I. Reid, H. Saito, and M.-H. Yang, Eds. Cham: Springer International Publishing, 2015, pp. 241–258.

[16] M. Zhang and Z. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1819–1837, Aug. 2014.

[17] G. Tsoumakas, I. Katakis, and I. Vlahavas, *Mining Multi-label Data*. Boston, MA: Springer US, 2010, pp. 667–685.

[18] K. Sozykin, S. Protasov, A. Khan, R. Hussain, and J. Lee, "Multi-label class-imbalanced action recognition in hockey videos via 3d convolutional neural networks," in *19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, June 2018, pp. 146–151.

[19] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757 – 1771, 2004.

[20] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," in *Machine Learning and Knowledge Discovery in Databases*, W. Buntine, M. Grobelnik, D. Mladenić, and J. Shawe-Taylor, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 254–269.

[21] J. Fürnkranz, E. Hüllermeier, E. Loza Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Machine Learning*, vol. 73, no. 2, pp. 133–153, Nov 2008.

[22] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *Machine Learning: ECML 2007*, J. N. Kok, J. Koronacki, R. L. d. Mantaras, S. Matwin, D. Mladenič, and A. Skowron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 406–417.

[23] Y. Peng, G. Chen, M. Xu, C. Wang, and J. Xie, "Multi-label learning by exploiting label correlations with lda," in *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, Nov 2017, pp. 168–174.

[24] Y. Guo and S. Gu, "Multi-label classification using conditional dependency networks," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, no. 1, 2011, pp. 1300–1305.

[25] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern Recogn.*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007.

[26] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, ser. NIPS'01. Cambridge, MA, USA: MIT Press, 2001, pp. 681–687.

[27] S. Zhang, Z. Li, J. Nie, L. Huang, S. Wang, and Z. Wei, "How to record the amount of exercise automatically? a general real-time recognition and counting approach for repetitive activities," in *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Dec 2016, pp. 831–834.

[28] H. Aloulou, M. Mokhtari, T. Tiberghien, R. Endelin, and J. Biswas, "Uncertainty handling in semantic reasoning for accurate context understanding," *Knowledge-Based Systems*, vol. 77, pp. 16 – 28, 2015.

[29] R. Mojarad, H. Kordestani, and H. R. Zarandi, "A cluster-based method to detect and correct anomalies in sensor data of embedded systems," in *24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, Feb 2016, pp. 240–247.

[30] R. Mojarad and H. R. Zarandi, "Two effective anomaly correction methods in embedded systems," in *CSI Symposium on Real-Time and Embedded Systems and Technologies (RTEST)*, Oct 2015, pp. 1–6.

[31] G. F. Cooper and E. Herskovits, "A bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, no. 4, pp. 309–347, Oct 1992.

[32] Z. Ji, Q. Xia, and G. Meng, "A review of parameter learning methods in bayesian network," in *Advanced Intelligent Computing Theories and Applications*, D.-S. Huang and K. Han, Eds. Cham: Springer International Publishing, 2015, pp. 3–12.

[33] Y. Zhao, R. Yang, G. Chevalier, X. Xu, and Z. Zhang, "Deep residual bidir-lstm for human activity recognition using wearable sensors," *Mathematical Problems in Engineering*, vol. 2018, pp. 1–13, 12 2018.

[34] E. Garcia-Ceja, C. E. Galván-Tejada, and R. Brena, "Multi-view stacking for activity recognition with sound and accelerometer data," *Information Fusion*, vol. 40, pp. 45–56, 2018.

[35] H. Sagha, S. T. Digumarti, J. del R. Millán, R. Chavarriaga, A. Calatroni, D. Roggen, and G. Tröster, "Benchmarking classification techniques using the opportunity human activity dataset," in *IEEE International Conference on Systems, Man, and Cybernetics*, Oct 2011, pp. 36–40.

[36] F. Li, K. Shirahama, M. Nisar, L. Köping, and M. Grzegorzek, "Comparison of feature learning methods for human activity recognition using wearable sensors," *Sensors*, vol. 18, no. 2, p. 679, 2018.